CLAIMS

What is claimed as new and desired to be protected by Letters Patent of the United States is:

1. A method for handling plug-and-play events occurring at a client, said method comprising the steps of:

(a) providing a client communicating with a server over a network using a presentation-level protocol;

(b) detecting a plug-and-play event notification regarding a device in communication with the client;

(c) redirecting said event notification to the server from the client: and (d) receiving, from the server, a command directed to said device.

2. The method of claim 1 wherein step (c) further comprises the steps of:

(c-1) generating a context identifier, said context identifier representing a virtual COM port;

(c-2) binding the context identifier to the event notification; and

(c-3) transmitting the bound context identifier and event notification to the server.

3. The method of claim 1 wherein step (c) uses a virtual channel to redirect said event notification.

4. The method of claim 1, wherein step (d) further comprises the steps of:

(d-1) receiving from a server a command including a generated context identifier;

(d-2) identifying the device using the context identifier; and

(d-3) issuing a command to the identified device.

5. The method of claim 1 wherein said event notification is generated as a result of a device arrival.

6. The method of claim 5 wherein said command is an open command.

7. The method of claim 1 wherein said event notification is generated as a result of a device removal.

8. The method of claim 7 wherein said command is a close command.

9. The method of claim 1 wherein said event notification is associated with at least one of a GUID, vendor ID, product ID and actual device name.

10. The method of claim 1 wherein the device in communication with the client uses one of the USB(Universal Serial Bus) protocol, IEEE 1394 protocol, Bluetooth protocol, wi-fi protocol, wireless protocol, and infrared (IR) protocol to communicate with the client.

11. A method for handling plug-and-play events occurring at a client in communication with a server using a presentation-level protocol, said method comprising the steps of:

a) receiving from said client a plug-and-play event notification regarding a device in communication with the client;

b) notifying an application program hosted by the server of the occurrence of the event notification;

c)receiving from the application program hosted by the server a command directed to the device; and

d) transmitting to the client a command directed to the device.

12. The method of claim 11 wherein the event notification received in step (a) from the client is received over a virtual channel.

13. The method of claim 11 wherein the event notification received in step (a) includes a context identifier bound to the event notification, said context identifier representing a virtual COM port.

14. The method of claim 11, further comprising the steps of:

creating a server-unique name to identify the device connected to the client that generated the event notification, said server unique name used in mapping the client device to a specific session on the server established by the presentation level protocol.

15. The method of claim 11 wherein step (b) further comprises the step of:

transmitting the event notification to applications communicating with the server within the session.

16. The method of claim 11 wherein step (b) further comprises the step of:

transmitting the event notification only to applications communicating with the server which have previously registered a callback for a type of event causing the event notification.

17. The method of claim 11 wherein said event notification is generated as a result of a device arrival.

18. The method of claim 17 wherein said command is an open command.

19. The method of claim 11 wherein said event notification is generated as a result of a device removal.

20. The method of claim 19 wherein said command is a close command.

21. The method of claim 11, comprising the further steps of:
detecting a break in the presentation level connection;
emulating on the server, in response to the detection of the break, a removal notification for the device connected to the client about which the event notification was received; and

saving a collection of session data for a user session established with said presentation level connection.

22. The method of claim 21, comprising the further steps of:

detecting a subsequent presentation level connection from a user; and

restoring the saved collection of session data to a session established with said presentation level connection.

23. A method for handling events occurring at a client in communication with a server using a presentation-level protocol, said method comprising the steps of:

a) receiving from said client an event notification regarding a device in communication with the client;

b) notifying an application program hosted by the server of the occurrence of the event notification;

c)receiving from the application program hosted by the server a command directed to the device; and

d) transmitting to the client a command directed to the device.

24. A method for informing a server about the presence of devices connected to a client, said method comprising the

steps of:

(a) providing a client communicating with a server over a network using a presentation-level protocol;

(b) emulating a plug-and-play event notification regarding a device in communication with the client;

(c) redirecting said emulated event notification to the server over a network ; and

(d) receiving, from the server, a command directed to said device.

25. The method of claim 24 wherein step (c) further comprises the steps of:

(c-1) generating a context identifier, said context identifier representing a virtual COM port;

(c-2) binding the context identifier to the emulated event notification; and

(c-3) transmitting the bound context identifier and emulated event notification to the server.

26. The method of claim 24 wherein the redirection of the emulated event notification in step (c) uses a virtual channel.

27. The method of claim 24, wherein step (d) further comprises the steps of:

(d-1) receiving from a server a command identifying the generated context ID;

(d-2) identifying the device using the context; and

(d-3) issuing a command to the identified device.

28. The method of claim 27 wherein the emulated event notification received in step (b) from the client is received over a virtual channel.

29. The method of claim 27 wherein the emulated event notification received in step (b) includes a context ID bound to the emulated event notification.

30. The method of claim 27 wherein step (c) further comprises the step of:

broadcasting the emulated event notification to applications communicating with the server.

31. The method of claim 27 wherein step (c) further comprises the step of:

transmitting the emulated event notification only to

applications communicating with the server which have previously registered a callback with the server for a type of event causing the emulated event notification.

32. The method of claim 24, wherein said client is a proxy client on a server, said server interfaced with at least one additional server.

33. A method for informing a server about the presence of network resources connected to a proxy client, said method comprising the steps of:

a) emulating a plug-and-play event notification regarding a network resource in communication with the proxy client;

b) redirecting said emulated event notification to a server; and

c) receiving, from the server, a command directed to said network resource.

34. An article of manufacture having embodied thereon computer-readable program means for handling plug-and-play events occurring at a client communicating with a server over a network using a presentation-level protocol, the article of manufacture comprising:

computer-readable program means for detecting a plug-and-play event notification regarding a device in communication with the client;

computer-readable program means for redirecting said event notification to the server; and

computer-readable program means for receiving, from the server, a command directed to said device.

35. The article of manufacture of claim 34 wherein the computer-readable program means for redirecting said event notification further comprises :

computer-readable program means for generating a context identifier, said context identifier representing a virtual COM port;

computer-readable program means for binding the context identifier to the event notification; and

computer-readable program means for transmitting the bound context identifier and event notification to the server.

36. The article of manufacture of claim 34 wherein the computer-readable program means for redirecting said

event notification uses a virtual channel to redirect said event notification.

37. The article of manufacture of claim 34 wherein the computer-readable program means for receiving, from the server, a command directed to said device, further comprises :

computer-readable program means for receiving from a server a command including the generated context identifier;

computer-readable program means for identifying the device using the context identifier; and

computer-readable program means for issuing a command to the identified device.

38. The article of manufacture of claim 34 wherein said event notification is generated as a result of a device arrival.

39. The article of manufacture of claim 38 wherein said command is an open command.

40. The article of manufacture of claim 34 wherein said event notification is generated as a result of a device removal.

41. The article of manufacture of claim 40 wherein said command is a close command.

42. The article of manufacture of claim 34 wherein said event notification is associated with at least one of a GUID, vendor ID, product ID and actual device name.

43. The article of manufacture of claim 34 wherein the device in communication with the client uses one of the USB(Universal Serial Bus) protocol, IEEE 1394 protocol, Bluetooth protocol, wi-fi protocol, wireless protocol, and infrared (IR) protocol to communicate with the client.

44. An article of manufacture having embodied thereon computer-readable program means for handling plug-and-play events occurring at a client communicating with a server over a network using a presentation-level protocol, comprising :

computer-readable program means for receiving from said client a plug-and-play event notification regarding a device in communication with the client;

computer-readable program means for notifying an application program hosted by the server of the occurrence

of the event notification;

computer-readable program means for receiving from the application program hosted by the server a command directed to the device; and

computer-readable program means for transmitting to the client a command directed to the device .

45. The article of manufacture of claim 44 wherein the event notification is received over a virtual channel.

46. The article of manufacture of claim 44 wherein the received event notification includes a context identifier bound to the event notification, said context identifier representing a virtual COM port.

47. The article of manufacture of claim 44, further comprising:

computer-readable program means for creating a server-unique name to identify the device connected to the client that generated the event notification, said server unique name used in mapping the client device to a specific session on the server established by the presentation level protocol.

48. The article of manufacture of claim 44 wherein the computer-readable program means for notifying an application program hosted by the server of the occurrence of the event notification further comprises:

computer-readable program means for transmitting the event notification to applications communicating with the server within the session.

49. The article of manufacture of claim 44 wherein the computer-readable program means for notifying an application program hosted by the server of the occurrence of the event notification further comprises:

computer-readable program means for transmitting the event notification only to applications communicating with the server which have previously registered a callback for a type of event causing the event notification.

50. The article of manufacture of claim 44 wherein said event notification is generated as a result of a device arrival.

51. The article of manufacture of claim 50 wherein said command is an open command.

52. The article of manufacture of claim 44 wherein said event notification is generated as a result of a device removal.

53. The article of manufacture of claim 52 wherein said command is a close command.

54. The article of manufacture of claim 44, further comprising :

computer-readable program means for detecting a break in the presentation level connection;

computer-readable program means for emulating on the server, in response to the detection of the break, a removal notification for the device connected to the client about which the event notification was received; and

computer-readable program means for saving a collection of session data for a user session established with said presentation level connection.

55. The article of manufacture of claim 44, further comprising:

computer-readable program means for detecting a subsequent presentation level connection from a user; and

computer-readable program means for restoring the saved collection of session data to a session established with said presentation level connection.

56. An article of manufacture having embodied thereon computer-readable program means for handling events occurring at a client communicating with a server over a network using a presentation-level protocol, comprising:

computer-readable program means for receiving from the client an event notification regarding a device in communication with the client; computer-readable program means for notifying an application program hosted by the server of the occurrence of the event notification;

computer-readable program means for receiving from the application program hosted by the server a command directed to the device; and

computer-readable program means for transmitting to the client a command directed to the device .

57. An article of manufacture having embodied thereon computer-readable program means for informing a server about the presence of devices connected to a client communicating with a server over a network using a presentation-level protocol, comprising:

computer-readable program means for emulating a

plug-and-play event notification regarding a device in communication with the client;

computer-readable program means for redirecting said emulated event notification to a server ; and

computer-readable program means for receiving, from the server, a command directed to said device.

58. The article of manufacture of claim 57 wherein the computer-readable program means for redirecting said emulated event notification further comprises:

computer-readable program means for generating a context identifier, said context identifier representing a virtual COM port;

computer-readable program means for binding the context identifier to the emulated event notification; and

computer-readable program means for transmitting the bound context identifier and emulated event notification to the server.

59. The article of manufacture of claim 57 wherein the redirection of the emulated event notification uses a virtual channel.

60. The article of manufacture of claim 57, wherein the computer-readable program means for transmitting the bound context identifier and emulated event notification to the server further comprises:

computer-readable program means for receiving from a server a command identifying the generated context ID;

computer-readable program means for identifying the device using the context; and

computer-readable program means for issuing a command to the identified device.

61. The article of manufacture of claim 60 wherein the emulated event notification is received over a virtual channel.

62. The article of manufacture of claim 60 wherein the emulated event notification includes a context ID bound to the emulated event notification.

63. The article of manufacture of claim 60 wherein the computer-readable program means for identifying the device using the context further comprises:

computer-readable program means for broadcasting the

emulated event notification to applications communicating with the server.

64. The article of manufacture of claim 60 wherein the computer-readable program means for identifying the device using the context further comprises:

computer-readable program means for transmitting the emulated event notification only to applications communicating with the server which have previously registered a callback with the server for a type of event causing the emulated event notification.

65. The article of manufacture of claim 57, wherein said client is a proxy client on a server, said server interfaced with at least one additional server.

66. An article of manufacture having embodied thereon computer-readable program means for informing a server about the presence of network resources connected to a proxy client, comprising:

computer-readable program means for emulating a plug-and-play event notification regarding a network resource in communication with the proxy client;

computer-readable program means for redirecting said emulated event notification to a server; and

computer-readable program means for receiving, from the server, a command directed to said network resource.

67. A method for enumerating devices communicating with a client that have been mapped into a session on a server, said method comprising the steps of:

launching an application in a user session on a server

intercepting device enumeration methods in the server-based user session;

redirecting the device enumeration methods to the server;

emulating an arrival event for at least one device enumerated by the redirected method, said device being a device in communication with a client system that was mapped into the user session prior to said application launch; and

notifying said application hosted by the server of the occurrence of the event notification.

68. A method for handling plug-and-play events occurring at a client, said method comprising the steps of:

(a) detecting a plug-and-play event notification regarding a device communicating with the client via a USB connection on the client;

(b) redirecting said event notification to a server over a network; and

(c) receiving, from the server, a command directed to said device.

69. The method of claim 68 wherein said event notification is generated as a result of a device arrival.

70. The method of claim 69 wherein said command is an open command.

71. The method of claim 68 wherein said event notification is generated as a result of a device removal.

72. The method of claim 71 wherein said command is a close command.

73. The method of claim 68 wherein said event notification is associated with at least one of a GUID, vendor ID, product ID and actual device name.

74. The method of claim 68 wherein step (b) further comprises the steps of:

(b-1) generating a context identifier, said context identifier representing a virtual COM port;

(b-2) binding the context identifier to the event notification; and

(b-3) transmitting the bound context identifier and event notification to the server.

75. The method of claim 68 wherein step (b) uses a virtual channel to redirect said event notification.

76. The method of claim 68, wherein step (c) further comprises the steps of:

(c-1) receiving from a server a command including a generated context identifier;

(c-2) identifying the device using the context identifier; and

(c-3) issuing a command to the identified device.